

Report on Lightweight Digital Collections Platforms

CUL Library Outside the Library

July 2014

Amanda Bartley
Dan Blackaby
Mickey Casad
Adam Chandler
Liza Flum
Emily Oliver
Melissa Wallace

Executive Summary

Our team started with a curiosity about Scalar and along the way formed a collaboration with two graduate students to compare it with three similar services. These “lightweight digital collection platforms” as we call them each have potential to fit the needs of students, staff, and faculty who want to build their own digital collection. Supporting this kind of scholarship using these tools today is very challenging, however, because none of them quite meet all of the requirements. In the end, we found ourselves imagining a drupal-based cloud platform and service designed for small digital collections, which we call “drupal.edu”.

Problem Statement

In 2013, the CUL Library Outside the Library committee designated a subgroup of members to investigate a range of easy-to-use content management systems. The group’s charge was to learn more about existing out-of-the box platforms, testing their suitability for library projects, applications, and audiences, with the aim of making recommendations to CUL for further investigation and exploration.

Building library capacity to support patron and staff use of lightweight content management systems could have a number of positive impacts.

Reduced dependence on web development staff resources

Currently, library web design staff create all web portals for digital collections and online exhibitions, managing a prodigious workload of continually creating new sites while updating existing ones.¹ Library web design services are always in demand. At the same time, there is a demonstrated desire among CUL librarians and curators for more direct creative and editorial access to online content management. From the perspective of curators, access to an online exhibit’s CMS could permit simpler updates and more responsive communication with donors, who frequently request webpage updates that may not make it into the web design team work queue. From the perspective of librarians, being able to create rich online content without enlisting the web development team would simplify planning for

¹ Some of the web technologies and software that are currently in use at CUL include Drupal, Project Blacklight (Ruby on Rails), DSpace, DLXS, Hydra, DPubS, Luna Insight, ARTstor/Shared Shelf, Kaltura (streaming video), bepress, Greenstone, Veridian, Confluence and WordPress.

collections-focused library events like physical exhibits or special classes. There is a strong sense among CUL librarians and curators that being able to support patrons or classes in using online exhibit environments would increase the visibility and the usability of the library's digital collections, and also move CUL more into line with the digital humanities support programs of peer institutions. We are also aware of a need for flexible platforms to create and readily adapt content for single-use situations like library classes.

Caveats and considerations

At Cornell, many kinds of online content management are already supported, not by CUL but by the Cornell University Academic Technologies unit, which provides technological support for classroom teaching. AT supports Blackboard and scholarly uses of Wordpress. The library therefore would have a slightly different niche to fill on campus, and will need to define this niche in a way that is not redundant. One pathway to this could be to understand CMS support as being an extension of the library's mission to improve scholarly access to library resources, and privilege systems designed for scholarly content that draws extensively on digital collections. Many content management systems include plug-ins and widgets with explicitly scholarly applications.

We considered a large set of possible platforms for our investigation, ranging from LibGuides, a familiar support for library content and library classes, to Blackboard, the educational content management program used most commonly across Cornell University, to EdX, the platform selected for Cornell University's first MOOCs. LibGuides proved so specialized in their usage that they didn't provide the breadth of uses we imagined. In a similar vein, Blackboard is useful and powerful for educational purposes, but very specialized in its uses and audiences, and furthermore already well administered on the Cornell campus by Academic Technologies. LibGuides and Blackboard both provided useful points of reference for our study, but did not fit the bill of our investigation.

Ultimately, we investigated the following platforms:

- Scalar, an award-winning up-and-coming CMS that emphasizes nonlinear organization of rich content
- Omeka, in some ways the 'industry standard' for collections-oriented scholarly content management and online exhibitions
- Drupal, a relatively easy to use open-source CMS
- Wordpress.com, a popular and very flexible blog engine
- Wordpress hosted by Cornell, a slightly more stable variation of the Wordpress engine with a wide menu of specifically educational plug-ins and add-ons.

Our investigation considered a wide range of criteria, documented in the appendix to this report, "Content Management Comparison". We attempted to consider qualities of relevance to the non-specialist user (librarian, curator, student, or library patron) as well as

the institution (university library and IT/repository support staff). In particular, noting that CUL must be mindful of the long-term sustainability and flexibility of any platform it supports, however casually, we focused on several aspects of a content management system's accessibility, longevity, and sustainability:

- Copyright considerations - whether content may be password-restricted
- Portability- whether an entire site may be moved to another platform without excessive intervention
- Hosting - what kinds of guarantees a platform made about content preservation
 - o This could include information about backup or 'Save API' features
- Cost - for example, membership or hosting fees
 - o This could include support for free or inexpensive offsite hosting options
- Design features - we also considered design features applicable to our scholarly audience, including:
 - o Responsive design
 - o Linked data / API availability
 - o Plug-ins and widgets that offer unique affordances

Literature Review

Articles on this topic are few and far between. Digital Forsyth, a digital library project involving the Forsyth County Public Library, Winston Salem University, and Wake Forest University decided to use Wordpress.com as a solution to their goal of community participation and user contributed content. Although their project required modifications to the Wordpress software including downloading plug-ins, modifying existing, and creating new functions, they found that Wordpress' blogging software best fit the project's need for a public interface, and found overall that the development of the user interface was possible, with the community-centric features that serve as the centerpiece of the project site (Mitchell 2008, p. 9).

The Cleveland Colby Archives at Colby-Sawyer College (NUASC) also uses Wordpress.com's website structure as part of their digital archives, Haystack, using the site's format to create a "this day in history" type blog by transcribing a page out of the diary of a former Hollywood actress corresponding to a given date (Bogan 2011). The Norwich University Archives and Special Collections had success utilizing Wordpress downloadable blogging software in creating a digital "catablog." They found this to be a good fit because the library was already familiar with using Wordpress for other projects and because of its ability to be customized to fit the project with minimal support from Norwich's IT department (Bogan 2011).

The Metropolitan New York Library Council found Omeka most appropriate for its digitalMETRO project. Despite Wordpress' ease of use and multiple plug-ins, they found that it didn't support the types of collection-building workflows and metadata-creation common to archives and libraries (Kucsma, Reiss, and Sidman 2010, p. 2). Finding the installation and customization of Omeka to be on the level of usability as Wordpress, METRO found Omeka to be an efficient platform for small to medium sized institutions to make online digital content available.

Allison March has used Omeka for the past three years (out of an anticipated 10 as an ongoing project) as a pedagogical tool in a graduate seminar, despite difficulties by her students navigating the CMS herself. Her students were assigned to create an online exhibit from a virtual object database as a lesson on database architecture. While the student completed projects themselves were "mostly disastrous," she nonetheless considered the experiment successful in introducing her students to the type of technical knowledge required to perform in digital humanities.

Scalar is a relatively new platform geared towards long-form, digital publishing (scalar webpage). Developed by the Alliance for Networking Visual Culture at USC, with extensive support from the Mellon Foundation. The platform has official partnerships with numerous archives, libraries, and university presses. Steve Anderson discusses Scalar's capacity of two way linking between the contents of electronic archives and their treatment in scholarly publications, for instance, the sharing of data sets created by his Technologies of Cinema archive (Anderson 2013).

Platforms in CUL Context

Content platforms

In designing our investigation of the select platforms listed above, we were mindful of the criteria and considerations that shape the development of online exhibits and public web portals at CUL. We also took into account the bigger picture and trajectory of digital collections and interface development at CUL. In addition to asking basic questions about a platform's simplicity of use and visual / multimedia capabilities, we reviewed such criteria as:

- flexibility of metadata profile
- ease of integration with digital repositories already supported at CUL
- capacity for batch import and export of files and metadata
- whether or not a platform dynamically reshapes itself for a mobile interface
- long term sustainability questions regarding hosting, backup, and future migration of content

Our full assessment matrix is available in the Appendix to this report.

Omeka and Omeka.net

Developed at the Roy Rosenzweig Center for History and New Media at George Mason University and initially released in 2008, Omeka has developed a strong reputation in museum and library communities as an easy to use platform for online collections and exhibitions. The platform is especially appropriate for static visual content. Omeka has been adopted by cultural heritage and educational institutions both large (NYU) and small as a platform for low-cost flexible exhibit design. There is a large online community of technical and context-focused support for Omeka (including, for example, suggestions for how to use Omeka in different kinds of educational or exhibition settings), which further lowers the entry cost for a new user. There is also a wide range of plug-ins and add-ons available for download or further development; Omeka is fully open source and officially encourages experimentation from its user group.

The full version Omeka is available for free download to any person or institution with a Linux-based Apache server (complete system requirements are available at [http://omeka.org/codex/Preparing to Install](http://omeka.org/codex/Preparing_to_Install)). For users without server access, there is a web-hosted version of the Omeka platform, Omeka.net, which includes a similar interface and many of the same features as the full server-based version of the platform, but entails a narrower range of plug-in and file size / file type options. For the sake of simplicity, our investigation focused on the web-hosted version, Omeka.net.

WordPress

WordPress is a content management system that started as an easy way to create blogs, but has grown to become a popular choice for larger and more complex websites. WordPress is available free through wordpress.com, or can be manually installed and configured, with greater flexibility, through wordpress.org.

WordPress includes most basic websites features, such as search, navigation, page creation, user and permissions management, and publishing workflow. There are also many plugins for adding more complex features (e.g. slideshows, share buttons, Google Analytics integration), some of which are free and some for a fee.

WordPress has a relatively low learning curve when managed solely through the browser-based administrative interface. However, you are somewhat limited in that interface depending on what themes you choose; some themes allow you basic

customization (fonts, colors) and/or access to templates and code through the browser, while others do not allow customization at all.

Cornell University Library (CUL) has a small number of websites built in WordPress. These sites are primarily blogs, containing news or informational posts, and the primary function for most are to meet the communication and outreach needs of various departments. Most of CUL's WordPress sites are hosted by the [Cornell University Blog Service](#). As such, these sites are relatively straight-forward and have not entailed much, if any, complex or custom development. Most sites use themes provided by Cornell University Blog Service, or other out-of-the-box themes from WordPress, and are fully administered through WordPress's dashboard in the browser. In relation to other content management platforms or software applications, WordPress's use at CUL is relatively low.

CUL WordPress sites include [CUL Discovery & Access](#), [DSPS Press](#), [problemsolved](#), [Cornell University Library Conservation](#), [History at CUL](#), and [@ Olin & Uris Libraries](#).

Scalar

Scalar bills itself as an “open source content management tool for born-digital, media-rich scholarly publishing.” It is targeted toward authors of born-digital scholarship who want to create media-rich websites online. Scalar supports multiple media types and allows authors to create rich data connections between these different media types. Like Wordpress and Omeka, Scalar is available download on a local server or hosted on the by USC, the sponsoring organization for Scalar.

Scalar seems especially geared toward essay- and book-like work, or work with a linear progression. It is relatively easy to set up through its browser-based interface and provides a choice of themes. Users can access a theme's CSS though the browser for basic user interface (UI) customization.

Scalar also has an API to allow for more complex customization and development. Scalar's website includes many examples of out-of-the-box websites as well as fully customized, highly interactive sites.

Thus far, Scalar has had minimal use at CUL, although interest in it as a platform to support digital humanities has grown in the past year. CUL's first investigations of Scalar began in the summer of 2013, when the platform was introduced as part of a pilot Fellowship in Digital Scholarship for humanities graduate students. As part of this program, graduate fellows had brief workshops and tutorials in a variety of digital scholarship and scholarly publishing tools, with an aim to creating their own digital project by the end of the summer fellowship period. An overwhelming majority of the pilot group (5 out of 6 students) chose

to build their projects in Scalar. Of these five, three projects emphasized specific classroom teaching applications.

Scalar's position between the realms of digital archives and scholarly publishing also recently attracted the interest of Cornell's [Signale series](#), a partnership between CUL, Cornell University Press, and the Cornell Institute for German Cultural Studies. A small team of CU librarians and developers did cursory research into Scalar to see if it would be a viable platform for creating interactive multimedia companions to monographs published through the series. In the end, the team decided to build A/V companions to Signale books using Drupal, the same platform in which the Signale website is built. Drupal was selected over Scalar because of the ease of integration of the audio components into the existing UI (audio pieces will be linked from the book title on the Signale website and displayed within the same site and design, so as to create a seamless user interface and experience). In addition, the streaming audio files will be stored in CUL's instance of Kaltura, and we had already developed other Drupal sites where we integrated Kaltura videos into the website. The team believes it will be relatively quick and easy to replicate that work in the Signale site. While Scalar might have been a good choice with its a/v support and optimization for book content, Drupal was chosen because we will have a seamless UI with the existing Signale website and anticipation that development time will be much shorter.

Drupal

Drupal is a powerful open source content management platform that is popular in many libraries. It has an active and supportive development community, which contributes to its popularity. Drupal is built in PHP with a MySQL database.

Drupal includes core website functionality, such as search, navigation, page creation, image and file support, and user and permissions management. In addition, there are "modules" for more complex interface features such as slideshows, maps, complex menus, taxonomy management and multilingual interfaces.

Drupal's UI is built from a highly flexible template. Developers can use out-of-the-box themes, or hand-coded, completely customized designs. Drupal also includes many responsive themes, optimized for everything from smart phones to larger desktop screens.

Drupal has a relatively high learning curve. While a non-coder can build a Drupal site solely within Drupal's browser-based administrative interface, the real power comes from being able to access the server, hand-code templates and modify existing modules or create new ones. The learning curve is still relatively high for non-coders who plan to build a site purely through the browser admin, without ever touching any code.

In recent years, Drupal has been the primary choice for most CUL websites. CUL has been using Drupal since 2008 and has developed very good in-house expertise in the creation and maintenance of Drupal sites. In addition, since Drupal use is high among libraries and has an active development community, CUL designers and programmers have a number of resources for getting support in developing features and applications that are relevant to the larger library community.

Drupal websites at CUL are varied and diverse. Drupal is the main choice for department or unit library websites, including the main Library website and several unit library sites (Engineering, Physical Sciences, Fine Arts, Music). All new unit Library websites are developed in Drupal.

Drupal has also been used in support of digital collections. In many cases it is used for small websites that serve as the introduction to collections stored in image repositories like ARTstor/Shared Shelf or Luna Insight (John Reys, Divine Comedy, Cornell Collection of Antiquities). It has also been used as the primary place for storing smaller, text-based collections (Waguih Ghali) and has been used as the front-end for video collections stored in Kaltura, CUL's video streaming service (Efraim Racker). It has also been used to develop a one-off, interactive accompaniment to a print monograph from the Signale series (Warburg).

In addition, CUL has plans to further investigate using Drupal with various APIs or other data sources, including pulling images and data from Shared Shelf using the International Image Interoperability Framework (IIIF) and querying the solr index for the Library's new Blacklight catalog. CUL has also developed several modules that pull data from the Library's legacy Voyager catalog.

Media hosts

Our two case studies both require robust support for audio and video with large file sizes. For that reason, we found it necessary to separate the presentation of the content from the hosting of the media files. Our appendix includes a table with criteria.

There are a variety of free options for hosting audio materials. There are distinct advantages to both off and on-site hosting. The primary advantages of off-site hosting come in increased storage and bandwidth, and with the ever-increasing number of methods by which the content can be shared, either directly or via a social media platform.

One of the more popular options is SoundCloud (<http://www.soundcloud.com>) . SoundCloud is currently rather ubiquitous as an audio platform on the Internet. It allows the users their own archival pages, as well as allowing for embeddable players of single tracks. Under their

terms of service, users still own their audio, but give SoundCloud and its users a limited use license to “reimagine” or rework these tracks into new original creations.

The Internet Archive (<http://www.archive.org>) is another commonly used audio Internet storage medium. The Internet Archive is a non-profit organization. Audio stored there is done so under a Creative Commons License, in what is termed an “Open-Source” community archive. Unlike SoundCloud, it allows for variations in its storage mediums, allowing for streaming, Ogg Vorbis, and the use of .zip files. It also allows users to embed a player. Sharing to social media is not available.

Another common option for the storing of media is YouTube (<http://www.youtube.com>) or similar services such as Vimeo (<http://www.vimeo.com>) . The primary disadvantage of YouTube is its ubiquity and high visibility to search engines, notably Google. It is a fine platform for video, allowing many options in regard to privacy and sharing, but it is a system primarily designed to handle video. As such, audio needs to be treated as video, resulting in inelegant presentation.

Mixcloud (<http://www.mixcloud.com>) allows the user to upload audio in much the same way that SoundCloud does. The terms of use are explicit in their lack of guarantee of confidentiality, and state that MixCloud may or may not use the user’s submissions totally at their discretion. Practically, MixCloud functions as a cross between SoundCloud and radio services such as Spotify or Pandora. Its primary purpose would be to host podcasts.

Another similar option is Bandcamp (<https://bandcamp.com>) Bandcamp is geared toward independent musicians, and while it has many similar functions to SoundCloud and MixCloud, it features a heavier emphasis on the social media aspects of the service.

The most common option is to host the audio within a personal website. The primary concerns are storage and delivery mechanisms. Storage really only becomes a concern when dealing with large amounts of high quality video. The delivery mechanisms vary, but the various platforms (Drupal, Omeka, etc.) do offer ways to embed players within a site. One consideration that should be addressed is whether downloads of content are allowed. While this is usually easier to set up, it allows for much less control over the way the files are ultimately copied or used.

Case Studies

We were fortunate to make contact with two graduate students in the Cornell English department, whose digital projects provided nearly perfect and entirely uncontrived case studies for our investigation. Emily Oliver and Liza Flum joined the investigation team in the Spring semester of 2014, built test sites in each of the CMS platforms under investigation,

and reported back to us on their findings. Each student had her own project, and each project presented slightly different requirements, objectives, and starting conditions.

Case Study: Knox Writers' House (Emily Oliver)

Knox Writers' House² is an online archive that offers audio recordings of living U.S. poets reading their work. Emily Oliver and her collaborators began recording the poems in 2010. The collection focuses geographically on the American Midwest, and traces the route of the creators' road trips to find and record poets. The interrelationship between poetry and locality is an important theme to the site. The site includes audio recordings, interview texts, original drawings of the poets, and maps of their location.

The original version of Knox Writers' House was built in iWeb. Its offline version is currently hosted at Knox College in Galesburg, Illinois. The original project is not currently up to date or sustainable: to start with, its platform iWeb is no longer supported. IP and display release pose another potential problem with this project: Emily and her partners did not ask poets to sign audio release agreements at the time of recording, and would need to re-contact every poet in the project to secure permission to display the recordings and the poems.

Because of Knox Writers' House, Emily already had a great deal of skill and experience with web-based content management systems. She hoped to find an easy-to-use platform for the new version of her project, but wanted whatever would be best for the project itself—even if this meant she would have to hire a developer to build her site.

Emily describes her project and the CMS assessment experience as follows:

Description

[Knox Writers' House](http://knoxwritershouse.com) is digital audio map of contemporary American writing. Since 2010, this project has taken me to cities and towns all over the United States to collect recordings from the poets, essayists and fiction who live there. Writers read their own work and a 'best-loved' piece by writer they admire. These recordings sessions, which have typically taken place over the kitchen table at a writer's home, conclude with an interview about the place she or he lives.

Poets Monica Berlin, Bryce Parsons-Twesten and I built the original KWH on iweb, which was discontinued by Apple shortly thereafter. While the current site functions well enough to boast regular visitors and is a tool in college and high school teaching, it is difficult to maintain and expand. Housed on the server of a small liberal arts school, the site crashes frequently and can only be edited or added to from one computer. Since each page on the

² <http://knoxwritershouse.com>

current website is static, when we inevitably spell a writer's name wrong or make a typo in the title of a poem, we have to go in and manually change everywhere that name or title appears on our site. These time consuming maintenance issues put the project's longevity in peril.

With the expert guidance of Cornell's Library Outside the Library team, I made small scale demo sites on Omeka, Wordpress and Drupal in order to examine what would be the most suitable candidate onto which I could migrate the current KWH project.

Primary criteria for this project (Emily)

The essential components is that the new platform would allow for the site material to be presented and browsed in a variety of ways and that it could be edited from multiple computers. Other secondary criteria include:

- podcast / mobile site capability
- blog page embedded in website
- ability to be stylistically consistent with current site
- ease with audio and other multimedia
- site content be searchable and organizable by place, writer and poem / story title.
- random poem / story generator.
- the ability to embed map to organize writers on.
- embed audio files / multimedia files in about page, homepage.

Rating of platform options, from worst to best

Omeka.net, WordPress, Drupal

Although I was impressed with much about the platform, Omeka employs a tiered organization system, with items, collections and exhibits which made it difficult to display content in multiple ways. In my [test site](#), I made the basic unit, the 'item', each individual audio recording of a poem, story or essay. But in order to create an Omeka exhibit [page](#) (featuring poet Carl Phillips) that was stylistically similar to the corresponding page on the current KWH site, I had to add author art as a item as well. Then, in the 'browse by item' section header, visitors see *recording, recording, art, recording*, which looks disorganized. I experimented with making the writer the item and attaching art and recording files to that item but then, visitors couldn't browse by individual pieces of writing. I also wanted to hide the 'Dublin Core' metadata because it wasn't relevant to the KWH purpose. The inability to do this with ease contributed to the conclusion that Omeka seems more usable for a strict academic archive than for the Knox Writers House, which serves both general and scholarly users.

Wordpress was extremely simple to use. I was able create a clean-looking demo without tapping into the wealth of online forums I had needed for Omeka and Drupal. That being said, I felt like I was always working against Wordpress default blog format. Content can be displayed in either a 'post' or a 'static page,' which means most of the KWH would have to

be hard coded individual pages, similar to what I did in lweb. In addition, there seemed to be major limitations with multimedia. For example, unless I purchased or built a plug-in, clicking an audio file takes visitors to a new page, rather than the player displaying on the same page, resulting in one more step between the user and our content. Though there are workarounds to this problem, wordpress does not seem, in general, to have the structural integrity to make the my demo site scaleable to Knox Writers' House actual size.

Drupal was by far the most challenging platform to use initially. Like Omeka, it has its own structural framework with its drupal-specific terminology. Unlike Omeka, this structure, once I got my bearings, was extremely flexible in how segments of content could be repurposed elsewhere on the website. For example, I uploaded the transcript of poet Carl Phillips interview about living and writing St. Louis, MO as a 'block.' I could set this block to appear on Phillips' page and a general page with the interviews about St. Louis. If I wanted to create a special display for, say, National Poetry Month, this block could also appear in a 'view' in an altered position if I simply changed the settings. This database driven organization really solves my major problems with the current website. I didn't need to choose if the basic structural unit the recording or the writer because I could make a 'content type' for 'writer' and for 'recording' and relate them to each other.

Case Study: Open access library of poetry teaching resources (Liza Flum)

Our other student collaborator, Liza Flum, came to us with a project that was still in its early planning stages. Liza envisioned a website with video lessons and lesson plans from teaching poets, which could be used for poetry education K-12 schools that did not have active poetry programs. The site would involve large numbers of video and audio files, as well as lesson plan documents, and potentially other tools as well.

In contrast to Emily, who had concerns about unchecked proliferation of Knox Writers' House audio files, Liza actively wanted users to be able to download video files from her site, in order that they might be used in classrooms that lacked fast internet connections. Like Emily, Liza came to this project with a great deal of prior knowledge and digital literacy. She wanted to test out many different platforms before committing to one for her project, but had her own list of desired characteristics for the platform. It should:

- Facilitate browsing of content multiple ways (by grade level, subject, and Common Core standard)
- Include multiple files with each item (video and multiple text files)
- Stream videos and make full file available for download
- Display an image next to the link for each lesson
- Include a comments feature for each lesson
- Have a mechanism for users to submit videos, and for administrator to evaluate and edit submitted content before posting
- Create displays of featured content on home page

What follows is a description of the project and CMS testing experience in Liza's own words.

Description

My digital humanities project, called The Facing Page, is an open access library of video lessons on poetry. This project is designed to supplement existing poetry curricula at the high school level. In the specialized field of poetry, I believe educators are facing a problem of transmission. Although there are many practicing poets in universities and local literary communities, poetry is too often limited and circumscribed in the classroom. With national benchmarks like the Common Core, students are often taught to read poems for assessment, not enjoyment. Through the Facing Page, I'm asking artists and educators to create short videos that deepen high school students' engagement with existing high school poetry curricula. Their videos will provide supplemental lessons on commonly-taught poems, encouraging students to read for pleasure and creativity as well as analysis. By integrating videos from the Facing Page into their existing lesson plans, high school teachers receive many of the benefits of successful Poets in the Schools programs. From the perspective of high school teachers, these brief videos of "visiting poets" enrich classroom discussion, provide valuable arts education, and give poetry a human face. From the perspective of poets, this project provides a vehicle for transmission, allowing them to share their skill and enthusiasm with students in an accessible and impactful way.

In this early stage of the project, I have been collaborating with the Academic Technology Center at Cornell to create high-quality audio and video recordings for the site. The videographers who work on MOOCs at Cornell have been especially generous with their time and expertise, teaching me how to film lessons in Cornell's recording facilities.

Primary criteria for this project

To showcase the lessons in the Facing Page, I'm seeking a platform that will integrate streaming multimedia content with text and audio files. The Facing Page will be built around individual lessons on poetry. Each lesson page will include a five-minute video of an educator presenting a mini-lesson, as well as text files including supplemental questions and prompts that the teachers can use in class. To simplify storage, I plan to stream these lesson videos through an embedded Youtube player rather than upload full video files. I'd still like the lessons to be accessible for classrooms without

internet connectivity, so I will also make audio files of the full lessons available for download. Teachers who adapt these lessons for use in their own classrooms will be encouraged to discuss their experiences on the site. I'd like to include mechanisms for community engagement, so that teachers can comment on the lessons and even submit their own videos and lesson plans for publication. To make the content easily accessible, I will be tagging the lessons with Common Core's standards, grade level, contributor name, and subject matter, and the content must be browseable by all these categories. I hope to be able to add new browse options as the collections grow. I'd like to be able to showcase featured lessons on the Facing Page's homepage, highlighting lessons that correspond to relevant events, times of year, and classroom units. There are already several successful websites that include all these features, such as [The Teaching Channel](#).

Rating of platform options, from worst to best

Through our independent study with the Cornell library, Emily Oliver and I created demo sites on Omeka, Wordpress, and Drupal to determine which platform was suited our projects. Based on this testing, I found that Omeka.net was least suitable for the Facing Page. Wordpress would meet the needs of this project, and Drupal would be the most scalable solution in the long term.

Omeka.net meets most of the basic needs of the Facing Page. It accommodates multiple files for each lesson, allowing me to display a central lesson video and related files as one "item." With the use of plug-ins, I can stream videos directly from Youtube, and I can add supplemental text and audio files as additional item-level attachments. Omeka.net also offers a plug-in for user participation, allowing teachers to comment on lessons and submit their own content for possible inclusion in the collections. Based on these features, Omeka.net would be an adequate platform for displaying the Facing Page in its initial stages. However, Omeka.net's organizational structure does not easily accommodate the browse options I'd like for this project, and the rigid site organization would not allow the collections to grow in new directions.

To upload content to Omeka.net, users must upload files as individual items, and then group them into collections. Each item comes preloaded with metadata fields which I had little ability to customize. I found that that the metadata fields available were not a good fit for capturing the information I needed to display. When items include multiple files, it is difficult to accurately describe the content with the available metadata fields. These items are then grouped into collections, which gather related

items together for easy browsing. Items cannot be cross-posted into multiple collections. To approximate the browsing structure I would like for the Facing Page, I grouped items into collections by grade-level and subject. Then I tagged each lesson with a Common Core standard, creating an alternative avenue for accessing the lessons. Further browsing options in Omeka are possible but limited. To create additional browse options, I tried to create exhibits that showcased lessons by theme or subject. Although exhibits are attractive, they need to be built manually, and they are not an efficient way to create cross-sections of a large amount of content. While workable, Omeka's organization and display options were just too limited for this project.

The second platform I tested, Wordpress, would have been a good fit for the Facing Page. Wordpress supports all of my desired features, including streaming Youtube videos, multiple associated files per lesson, and interactive comments. Most importantly, Wordpress's flexible site organization allows for multiple pathways into the content. I was able to create multiple ways of browsing content by grouping lessons into several categories at once. Unlike Omeka, I could easily cross-post lessons into multiple categories. Categories can also be added or removed in bulk, allowing large-scale changes to the site organization. The tag feature creates another possible pathway into the lessons. I tagged each lesson with Common Core standards, so that several Common Core standards would display as hyperlinks on the lesson pages. Tags can be applied to or removed from posts in bulk, and tags can also be converted to categories, creating further flexibility in site structure. This flexible site organization is the most appealing feature of Wordpress, and it would allow me to expand and shift the structure of the Facing Page site as the collections grow.

In Wordpress, the process of adding and editing content at the item-level is sometimes cumbersome. Because Wordpress is built on a blogging platform, each video must be uploaded in blog post format. Through the use of html code, I added the relevant metadata to each video and uploaded a full-text document including lesson plans and an audio file. My main concern with Wordpress is that each post needs to be hard-coded, creating repetitive work when uploading lessons. Any changes to the display of the lesson pages would mean individual edits to each post throughout the site. Although categories and tags allow for flexible site organization, the lessons cannot be browsed by any of the metadata that is entered into the html page for each post. For example, even though I will be entering the poet's name into the text of each posted lesson, I cannot make the posts browseable by poet without adding each poet as a tag or category. These concerns mean that Wordpress, although certainly an adequate platform for the Facing Page, might be less scalable than other solutions.

When I think about the long term growth of this project, Drupal seems to be the best solution. Drupal meets all my requirements for the site: it can stream video files,

display multiple text and audio files for each page, and allow for comments and user submissions. Drupal also allows for the most flexibility in organizing, editing, and restructuring the site. To build a basic Drupal site, I created a content type for lessons. Each lesson I uploaded included a streaming video and multiple associated files. I used the view feature to create cross-sections of the lessons according to theme, grade level, and Common Core standard. I am drawn to Drupal because it will allow for the greatest flexibility in organizing the site, enabling me to break out lessons according to a variety of views. Drupal also allows me to easily update repeated information, like poet name and grade level. By editing blocks, I can update the information that appears alongside each lesson easily without re-entering text into HTML pages. .

A final consideration is ease of set-up. Drupal has a considerable learning curve. Although I have created a basic Drupal site, I recognize that my site will require custom design work before it has a presentable front-end. Meanwhile, Omeka is straight-forward to set up, but my demo of the Facing Page has substantial organizational and display limitations. Of all the platforms I tested, Wordpress was the easiest to set up and required the least amount of customization. In the short term, as I apply to grants, I am considering using Wordpress as a demo site to create a proof of concept for grant applications, which I work on building a long-term home for the Facing Page on Drupal.

Copyright

Copyright issues for sites such as these can be numerous. It is rare when using embedded multimedia that copyright for all content is held by a single individual. Consideration should always be given to the source of the content, the use to which the content is being put, and whether or not these uses qualify for a “fair use” exemption under copyright law. For a more extensive exploration of the topic, see Peter Hirtle, Emily Hudson and Andrew Kenyon’s book, *Copyright of Cultural Institutions*.³

A major factor is whether the content is created for the site itself, or was derived from outside sources. Created content is easily covered by a ‘blanket’ license statement. Using outside content requires at least two elements -- attribution and releases. Attribution is proper in any case, but especially if releases cannot be obtained. No media content should be posted without attribution, even if created for the site itself. This is the case even for embedded media which is hosted elsewhere.

³ Peter Hirtle, Emily Hudson, and Andrew T. Kenyon, *Copyright of Cultural Institutions: Guidelines for Digitization for U.S. Libraries, Archives and Museums*, Cornell University Library: Ithaca, New York (2009).

The copyright issues faced by our two case studies presented two very different scenarios. One project was being made from scratch, and thus had the freedom to use copyright in whatever method they preferred. They had the choice of being totally restrictive or totally open about how their content was used. The second project, however, consisted of migrating an extant site containing many audio files for which copyright clearance had not already been obtained. In audio recordings, copyright will depend on the relative contributions of the people creating the recording.⁴ Although all the recordings in this case are nominally done as part of a distinct project, primary copyright is held by the individuals being recorded. Although releases were not acquired at the time of the recording for inclusion on the site, they were acquired after the fact. (See appendix for release form used)

For both sites, we suggested the use of a Creative Commons license. (<https://creativecommons.org/licenses/by-nc-nd/4.0/>) . For migration site, we collaborated on the site's owner on a copyright release form, which she then distributed to the original copyright owners of each audio file. Given that this form incorporated the same license as suggested for the 'new' site, both project owners will use this form going forward.

Each of the test sites is geared primarily toward educational purposes. With that in mind, both authors applied Creative Commons licenses to the sites (<https://creativecommons.org/licenses/by-nc-nd/4.0/>). The site geared primarily toward audio recordings of poetry had the additional task of securing releases from authors. These releases included the caveat that any license granted by them does not allow for uses primarily intended or directed toward commercial advantage or monetary compensation. If users of the site would like to repurpose the material commercially, they need to consult the original authors.

Findings

1. Wordpress and Scalar did not fit the needs of these two case studies.
2. Omeka.net is the closest to being designed for these two use cases, but the metadata and user interface inflexibility could not be ignored. Furthermore, the business model is premised on institutional funding to pay its high cost of audio and video hosting on Omeka.net servers, which made it too expensive for these two students.
3. Drupal's flexibility is a great strength. With expert assistance and guidance from Melissa Wallace it was possible to the students up and running quickly with design and functionality that fit 80% of their needs. Getting sites like these to the 100% point

⁴ Ibid, p. 56.

would require either significant training for the students or some non-trivial amount of dedicated support that were not in a position to support in our investigation.

4. A recurring theme in during our analysis was how to appropriately define library support to students such as Emily and Liza, or adjunct or visiting faculty, who are here for a couple of years or more, but then move on. What we discovered during our study is a grey area between permanent Cornell hosted collections and commercial hosting services. Within the library, our focus is on building and preserving collections that are clearly a part of the Cornell institution, that is what we are setup to support. The complication is these projects may not be finished when they leave Cornell. Students like Liza and Emily want to continue to add to the collections they are building after they leave Cornell. The projects they want to develop are scholarship that deserves to be supported while they are at Cornell on platforms they can use after they leave.
5. Copyright consultation is needed for projects such as these two case studies. The platforms that are available, omeka.net, scalar, wordpress.com, do not offer this service. They are software only.

Platform Pros and Cons

Platform	Pros	Cons
Scalar	<ul style="list-style-type: none"> • free • easy to use • like a modern version of omeka, with better support for remotely hosted media files • exhibit-based • API available 	<ul style="list-style-type: none"> • book-centric orientation didn't have capabilities needed for projects • no mobile view • no/poor metadata support
Wordpress.com	<ul style="list-style-type: none"> • easy to use • mobile themes available 	<ul style="list-style-type: none"> • doesn't handle media the best • "expensive" beyond minimal content, for hosting video • quickly hit limitations on free version with a/v • too blog-focused • didn't support object-oriented approach • no/poor metadata support

Omeka.net	<ul style="list-style-type: none"> • out-of-the-box metadata • object-oriented • good categorization • supports collections • made for digital collections 	<ul style="list-style-type: none"> • best metadata support among platforms what we investigated, but rigid dublin core adherence is frustrating • very difficult to embed video; worked with youtube but does not support internet archive (will not links to internetarchive.org) • business model and design expects media files to be uploaded, but if you have large files the cost per month becomes expensive • no mobile themes
Drupal	<ul style="list-style-type: none"> • supportive/active development community • adaptable - can adapt to changing needs of site, fully customizable • object-based • customizable design/attractive • mobile themes available 	<ul style="list-style-type: none"> • high learning curve • limited in what you can do through the browser • no free cloud version (needs to be installed campus or ISP server) • may need advanced coding skills, needs to frequent updates

Recommendations

We have two recommendations. The first is short-term and easily achievable. The second is much larger in scope.

Outreach and education

Offering workshops in simple content management systems through the library would also help to address an ongoing challenge: educating the academic community about the breadth of digital tools and digital collections that the library supports. Offering CMS support and education through the library would be commensurable with the programs of many peer institutions. At CUL, this could fit into a program of outreach focused on the affordances

of digital scholarship that already includes workshops, consultation, and cost-recovery support in such areas as:

- o Online scholarly publishing
- o Working with digital collections
- o Tools for algorithmic data analysis and visualization
- o Copyright consultation and digitization

Support for an easy-to-use online content management system could be seen to be a missing piece in this digital scholarship program.

Build a Drupal-based academic cloud platform for students, staff, faculty, and alumni

Our study suggests that Drupal is the platform with the most potential to accommodate a wide range of small digital collection projects. The problem is it not offered as reasonable cost in an easy to use, turn-key hosted academic-oriented cloud service for academia like omeka.net and scalar. In the same way that omeka was built on wordpress, a service that fits the requirements of our two case studies and similar ones could be built on Drupal. We see this as a service that higher ed institutions would subscribe to on behalf of the students and faculty (all faculty, visiting, adjunct, and permanent). Let's called it "drupal.edu". Collections built on drupal.edu would continue to be supported after students and faculty leave the sponsoring academic institution. Hosting on drupal.edu would be a benefit for alumni of the institution. A service such as this is needed to fill the gap between the strict guidelines and high threshold we have websites hosted on our servers in the cornell.edu domain and throwing students out to fend for themselves among the thousands of commercial internet website hosting companies.

Our imagined service would include consultation from expert drupal designers/developers, a "special buddy" as one of our students called it. Copyright consultation would also be available. There would probably need to be a cap on the number of hours each customer receives; that kind of detail could be ironed out in the course of grant funded pilot with partners such CU Academic Technology Center, the LYRASIL organization, and another institution, perhaps Columbia, under the 2CUL umbrella. LYRASIL might be a good fit because they have a business model for supporting libraries in cooperative ventures, with a particular focus on open source software and content.⁵ LYRASIL is just one possible partner, used here as an illustration of the type of non-profit we need.

⁵ <http://www.lyrasis.org/about/Pages/default.aspx>

References

- Anderson, S. (2013, November 6). Public intellectuals and open data. *Technohistory.net*. Retrieved from <http://www.technohistory.net/blog/public-intellectuals-and-open-data/>
- Bogan, K. (2011). Creating a digital archives with WordPress. *Library Technology Reports*, 47(3). Retrieved from http://go.galegroup.com/ps/i.do?id=GALE%7CA255559918&v=2.1&u=nysl_sc_cornl&it=r&p=AONE&sw=w&asid=048772fb0f8be7325b39b676443a6fbc
- Darlack, James. (2013). Diktuon: Drupal - CMS and Beyond. *Theological Librarianship*, 6(2).
- Fenton, W. (2013). Scalar. *PCMAG*. Retrieved October 30, 2013, from <http://www.pcmag.com/article2/0,2817,2419697,00.asp>
- Jones, K. M. L., & Farrington, P.-A. (2011). *Using WordPress as a Library Content Management System: A Library Technology Report* (p. 65). ALA Editions.
- Kucsma, J., Reiss, K., & Sidman, A. (2010). Using Omeka to Build Digital Collections: The METRO Case Study. *D-Lib Magazine*, 16(3/4). doi:10.1045/march2010-kucsma
- Marsh, A. C. (2013). Omeka in the classroom: The challenges of teaching material culture in a digital world. *Literary and Linguistic Computing*, 28(2), 279-282. doi:10.1093/llc/fqs068
- Mitchell, E., & Gilbertson, K. (2008). Using Open Source Social Software as Digital Library Interface. *D-Lib Magazine*, 14(3/4). doi:10.1045/march2008-mitchell
- Morton, A. (2011). Digital Tools: Zotero and Omeka. *The Journal of American History*, 98(3), 952-953.
- Omeka versus Wordpress - Forums - Omeka. (n.d.). Retrieved January 24, 2014, from <http://omeka.org/forums/topic/omeka-versus-wordpress>
- Research librarians discuss new models. (2013). *Library Journal*, 138(7), 15.
- Rumsey, A. S. (2013). *Meeting on Scholarly Production and Authoring: From Projects to Communities of Practice*. Scholarly Communication Institute, University of Virginia

Library. Retrieved from

http://uvasci.org/wp-content/uploads/2013/.../Report_SCI-SPA_22MAY13.pdf

Scalar About. <http://scalar.usc.edu/scalar/>

Thomas, Melanie. (2012). The Florida Center for Library Automation Experience Using Drupal. *The Serials Librarian*, 63, 170-172.

WordPress › Support » Using WordPress for digital archives and repositories. (n.d.).

Retrieved January 24, 2014, from

<http://wordpress.org/support/topic/using-wordpress-for-digital-archives-and-repositories>

Wordpress About. <http://wordpress.org/about/>

Appendix

Content Management Comparison

criteria:	scalar	omeka	drupal	wordpress.com	wordpress at Cornell
	http://scalar.usc.edu/	http://www.omeka.net/	https://drupal.org/hosting	http://wordpress.com/	http://blogs.cornell.edu
editor/platform functionality					
create account	Yes.	Free “Basic” plan allows 1 site and 500 mb of storage space.	Yes	Y	yes
add homepage	Y	Editor UI is oriented around Items and Collections.	Y	Y	yes (can set & customize static front page)
add subpage	Y	Yes. Can choose Parent for every item.	Y	Y	yes
add text	Y	Yes. Simple.	Y	Y	yes

		Yes. Simple for uploading audio into site. best practice for linking to external audio file not clear. Omeka dropbox plugin might be a clever option. There is also a Soundcloud plugin: http://omeka.org/add-ons/plugins/beam-meup-to-soundcloud/ The group that developed this plug in is also working on something similar for Internet Archive! Not sure if these are available in Omeka.net though.			
add audio	Yes, through Scalar "media."		Y	No \ \$99 upgrade	Yes
add video	Yes, through Scalar "media."	Yes. Simple for uploading audio into site. best practice for linking to external audio file not clear.	Y	No \ \$99 upgrade	yes
add images	Yes, through the Scalar "media."		Y	Y	yes

add presentation	If this is just a link to a PPT file, yes.	Yes. They even have an item type called Lesson Plan.	Y	Y	Can embed; plugins to simplify this
add documents	Is this just links to Word docs and such? Then yes.	Yes.	Y	Y	yes
link between pages	Yes	Yes, though default is to put items in collections and navigate that way.	Y	Y	yes
setup navigation	Yes	Easy because it is pretty much all predefined.	Y	y	yes
change theme	Yes	Yes. There are at least five different themes. Changing site them is trivial.	Y	Y / \$79 upgrade	yes
add search feature	Yes	Includes search.	Y	Y	yes
collaboration permissions	Yes, though there seem to be limited roles (author, commenter, reviewer, editor).	Yes, Very robust, granular permissions for collaborators in site.	Yes. Permissions can be customized based on roles, which are unlimited.	yes	yes: administrator, editor, author, contributor, subscriber
customization potential and learning curve	Somewhat easy to set up, but there may be limited customization without use of the API. Also	Easy to get started. East to change themes, but only to those offered in the omeka.net plan one subscribes to.	Highly customizable, but requires some knowledge of web technologies , languages	Customize with \$\$\$ upgrade and/or wordpress.org	Easy customization with themes, and lots of themes to choose from. There's a custom CSS plugin, too, but this has a warning label about

	you may need to depend on USC for certain edits or fixes. I removed myself as “author” of my site, as the site implies it is a book, and I’m not the author. I could no longer edit my site after that. I had to contact Scalar to reinstate me as author so I could edit it again.	HTML is available to edit.	and familiarity with web servers.		requiring prior CSS facility.
metadata	Custom and Dublin Core available.	Omeka uses Dublin Core metadata elements for items. Lots of attention to capturing detailed metadata for item types. This might be the distinguishing feature that makes this a good choice for students.	Custom, or various modules for integrating various metadata schemas, and schema.org support	N	There’s a plugin that will import and display metadata from Creative Commons image and media library; other than that, not sure.

extensibility	API; unsure if there are plugins	List of plugins: http://actest.omeka.net/admin/plugins Note: More plugins are available in the higher priced plans.	Yes; large developer community	Plugins available	Lots of plugins. Lots of widgets. There seems to be an API hook page for creating your own plugins?
public interface					
mobile UX	Yes, though may be by theme selection and not default.	Theme called "seasons" is not responsible in mobile device. Not sure if that is tru of all themes in the free plan.	Yes, by theme	Yes	yes
public access?	y	y	y	y	y
batch					
import content		Yes for metadata and to a file. https://omeka.org/blog/plugin_categories/batch-importing/	Yes	Y	Can import an entire blog if it's on Edublog, Blogger, or WordPress. NO batch import of libraries that I can identify, but I asked the support line about this.
export content		Yes for metadata: omeka-xml or OAI-PMH, for example. Unclear how to batch export audio files.	Yes	Y	See above: can also export entire blog to a Edublog, Blogger, or WordPress target.

legal/financial					
host: individual or institutional	individual or institutional	individual may create a site for free.	individual or institutional	Individual	Supported and hosted at Cornell by CIT. The CU Blogs service is powered by Edublogs, which is an academic version of WordPress.
cost	free	Plans available from free to \$999/year. http://www.omeka.net/dashboard/upgrade-plan	free and paid	Free with \$\$\$ upgrades	free to Cornell community
terms of service	http://scalabr.usc.edu/terms-of-service/	http://info.omeka.net/sign-up-for-account/terms-of-service/	Licensing FAQ https://druupal.org/licensing/faq		Not clear. I wrote to the support line about this.
technical					
preservation					Not clear. I wrote to the support line about this.
security/authentication	?		Yes	Privacy options	<p>Can choose privacy settings, including password protection, but I don't entirely know what the terms of this are.</p> <ul style="list-style-type: none"> • Google Apps • LDAP • Shibboleth
API	Yes	output formats: atom · dcmes-xml · json · omeka-json ·	Yes	Y, see Jetpack Plugin at http://jetpack.me/	Not clear. I wrote to the support line about this.

		omeka-xml - rss2			
max file size	2MB for upload	Individual file size is limited to 64mb for files uploaded into Omeka.net.	No limit, but based on your own server or hosting company.	3GB upload space	18 MB for upload
open source code?	Yes	Software is open source. http://www.gnu.org/copyleft/gpl.html	Yes	Yes	?? I don't think so?

Audio Comparison

	Internet Archive	soundcloud	Pod-O-Matic	Bandcamp	Mixcloud
	https://archive.org/details/audio	https://soundcloud.com	https://www.podomatic.com	www.bandcamp.com	https://www.mixcloud.com
cost	free	https://soundcloud.com/pro	https://www.podomatic.com/path/signup	Free; They make money via revenue share on sales; so they expect you to price your uploads. Bandcamp Pro is \$10 per month; allows for batch uploading and your own domain.	free
terms of service	https://archive.org/about/terms.php	https://soundcloud.com/terms-of-use	http://www.podomatic.com/about/tos		http://www.mixcloud.com/terms/

preservation	https://archive.org/about/#storage				
linking to file	https://archive.org/help/audio.php?identifier=Testaudiofile_201312 . There is an audio poetry collection for poetry: https://archive.org/details/audio_poetry	https://soundcloud.com/poetryfoundation/poetry-to-fight-the-dark	http://rokecabrera.podomatic.com/		http://i.mixcloud.com/CFqwIQ
demo file	https://archive.org/details/Testaudiofile_201312	https://soundcloud.com/poetryfoundation/poetry-to-fight-the-dark	http://rokecabrera.podomatic.com/		http://i.mixcloud.com/CFqwIQ
storage	Apparently unlimited; negotiated directly with archive.org	Free: 5 tracks per month, any size; 10 most recent visible to users; 2 total upload hours, 100 downloads/sound Pro - \$6 per month or \$55 per year: :4 total upload hours, 1000 downloads/sound Pro Unlimited \$15 per month or \$135 per year: Unlimited upload hours Unlimited downloads; Spotlight feature Single sound files are limited to 2 GB and total sound upload time per week is limited to 30 hours	Free: 500 MB of storage; 15 GB of bandwidth Pro: 2 GB of storage; 100 GB of bandwidth	No apparent limitation	Only streams; no downloads; Treats uploads as radio shows No “pro” level”